

Creating a view with documents

Language: [Deutsch](#) · [English](#)

Creating a view with documents

Show documents from the file system in a view – via proxy BO, data source and server-side rule.

TUTORIAL

APPLICATION DEVELOPER

UPDATED: JUL 2026

APPLIES TO NUCLOS 4.2026.X

On this page

- [Why a proxy BO?](#)
- [1. Create the proxy BO](#)
- [2. Data source for the view](#)
- [3. Implement the interface in a rule](#)
- [4. Add to the layout](#)
- [Related pages](#)

Why a proxy BO?

A virtual or generic business object is not enough for a view that shows **documents**: the files are not in the database but in the server directory [nuclos-home]/data/documents. The database only holds name and ID.

Therefore we use a [proxy BO](#). This builds the view via **Java code** so we can load the documents from the file system.



Note

Proxy BOs can only be attached as a subform to a parent object. To make it standalone, you need a write proxy BO.

1. Create the proxy BO

Use the BO wizard to create a proxy BO with at least one field for the **document attachment** and a **reference to the parent object**. An interface is generated automatically, which we implement later.

Attribute bearbeiten

Fügen Sie Attribute hinzu, ändern Sie vorhandene oder löschen Sie welche. Die Reihenfolge der Attribute wird in Listen und Unterformularen als Standard für die Spalteneinstellungen verwendet.

Anzeigename	Beschreibung	Datentyp	Feldbreite	Nachkommastellen	Eindeutig?	Historie?	Pflichtfeld?	Feldname	Javatype	Berechnet	Defaultwert	Attributgruppe
ElternObjekt	ElternObjekt	Referenzfeld	255		☐	☐	☒	elternobjekt	String	☐		Grunddaten
Dokument	Dokument	Dokumentenanhang	255		☐	☐	☐	dokument	GenericObjectDocumentFile	☐		Grunddaten

Proxy BO in the wizard.

2. Data source for the view

Create a data source that provides these fields and can be restricted to the parent object via a parameter:

- `elternid` – reference to the parent object
- `dokumentid` – document ID (table `t_ud_documentfile`)
- `dokumentname` – document name (table `t_ud_documentfile`)

Name

Beschreibung

Regelrelevant? ☐

Modell SQL **Vorschau** Java Klasse generieren?

Statement wieder herstellen

Parameter...	Datentyp	Pflichtpara...
elternid	Long	☐

```

select
    --Referenz auf Eltern-Objekt
    a.intid,

    --Dokument-ID
    pd.struid_dokument,

    --Dokument-Name
    doc.strfilename,

    --Weitere Felder
    [...]

from elternobjekt a
--Weitere Joins
[...]
--Objekt mit Referenz auf Dokument
inner join tabellex x on [JOIN]
inner join t_ud_documentfile doc on doc.struid = x.struid_dokument
where [BEDINGUNGEN]
--Einschränkung auf Eltern-Objekt
and $elternid = a.intid

```

Data source of the view.

3. Implement the interface in a rule

Create a rule via *Rule sets* [server-side rules](#) (ideally named after the interface). For the standard case `getByElternObjekt(Long)` is enough – it is called when the proxy BO is shown via the parent object's layout. All other methods may return null.



Download from the proxy BO

If the document is to be downloaded directly from the proxy BO, you **must** also implement `getId()` – with a unique ID per document.

```

package org.nuclet.[nuclet];

import java.io.File;
import java.io.FileFilter;
import java.lang.reflect.InvocationTargetException;
import java.lang.reflect.Method;
import java.nio.file.Path;
import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;
import java.util.Map;

import org.apache.commons.io.filefilter.WildcardFileFilter;
import org.nuclos.api.common.NuclosFile;
import org.nuclos.api.exception.BusinessException;
import org.nuclos.api.provider.DatasourceProvider;
import org.nuclos.api.provider.FileProvider;

public class BeispielObjektProxyImpl implements org.nuclet.[nuclet].BeispielObjektProxy{
    /*
     * Getter-Methode für das Eltern-Objekt.

```

```

        * Wir nutzen den Parameter, um das Ergebnis der Datenquelle einzuschränken.
        * Es muss immer ein neues Proxy-Objekt erstellt werden, welches mit den Daten
        * aus der Datenquelle gefüttert wird.
        * Alternativ können die Daten auch von einer anderen Stelle kommen (Remote-Verbindung, QueryProvider
etc.).

        * Für den Dokumentenanhang holen wir uns mit Hilfe von dokumentid und dokumentname
        * über die Methode getDocument() die Datei aus dem Installationsverzeichnis.
        * @param pAngebotId = Id des Eltern-Objekts.
        * @return Alle Zeilen des Proxy-BOs, die im Eltern-Objekt angezeigt werden sollen.
        */
@Override
public List<BeispielObjekt> getByElternObjekt(java.lang.Long pElternobjektId) {
    List<BeispielObjekt> result = new ArrayList<>();
    /*
     * Setze das Verzeichnis, in dem die Datei zu finden ist.
     */
    File dir = new File(getHomeDir() + "/data/documents/");
    try{
        /*
         * Setze die Parameter für die Datenquelle und führe sie aus.
         */
        Map<String, Object> params = new HashMap<>();
        params.put("elternid", pElternobjektId);
        for (Object[] row : DatasourceProvider.run(BeispielDatenquelleDS.class, params).
getRows()){
            /*
             * Erstelle neues Proxy-BO und füttere es mit Daten.
             */
            BeispielObjekt b = new BeispielObjekt();
            Long elternid = (Long) row[x];
            b.setElternId(elternid);
            String dokumentid = (String) row[y];
            String dokumentname = (String) row[z];
            b.setDokument(getDocument(dokumentid, dokumentname, dir));
            /*
             * Setze weitere Felder
             */
            b.set...(..);

            result.add(b);
        }
    } catch (BusinessException e) {
        e.printStackTrace();
    }
    return result;
}

/*
 * Ermittelt das Nuclos-Installationsverzeichnis.
 */
private String getHomeDir() {
    String home = "";
    try {
        Class<?> clz = Class.forName("org.nuclos.server.common.NuclosSystemParameters");
        Method getNuclosHome = clz.getMethod("getNuclosHome");
        home = ((Path) getNuclosHome.invoke(null)).toString();
    } catch (IllegalAccessException | IllegalArgumentException
        | InvocationTargetException | ClassNotFoundException | NoSuchMethodException |
SecurityException e) {
        e.printStackTrace();
    }
    return home;
}

/*
 * Suche das Dokument im gegebenen Verzeichnis, packe es in eine NuclosFile und gib
 * dieser den gegebenen Namen (andernfalls würden die Dokumente in der View unter ihrer
 * ID angezeigt werden).
 * @param documentId = ID des Dokuments.
 * @param anzeigenname = Gewünschter Anzeigenname des Dokuments.
 * @param dir = Verzeichnis, in dem das Dokument gesucht wird.

```

```

    * @return Eine NuclosFile oder null, wenn kein passendes Dokument gefunden wurde.
    */
    private NuclosFile getDocument(String documentId, String anzeigename, File dir) throws
BusinessException {
        FileFilter fileFilter = new WildcardFileFilter(documentId + ".*");
        File[] files = dir.listFiles(fileFilter);
        if (files.length == 1) {
            NuclosFile file = FileProvider.newFile(files[0]);
            file.setName(anzeigename);
            return file;
        }
        return null;
    }

    /*
    * Alle anderen Methoden müssen nicht weiter implementiert werden.
    */

    @Override
    public void setUser(org.nuclos.api.User user) {
    }
    @Override
    public List<BeispielObjekt> getAll() {
        return null;
    }
    @Override
    public List<java.lang.Long> getAllIds() {
        return null;
    }
    @Override
    public BeispielObjekt getById(java.lang.Long id) {
        return null;
    }
    @Override
    public List<BeispielObjekt> getByDokument(org.nuclos.api.UID pNuclosdocumentfileId) {
        return null;
    }
    @Override
    public void insert(BeispielObjekt pweitereDokumente) throws org.nuclos.api.exception.BusinessException {
    }
    @Override
    public void update(BeispielObjekt pweitereDokumente) throws org.nuclos.api.exception.BusinessException {
    }
    @Override
    public void delete(java.lang.Long id) throws org.nuclos.api.exception.BusinessException {
    }
    @Override
    public void commit() {
    }
    @Override
    public void rollback() {
    }
}

```

4. Add to the layout

Finally, add the proxy BO to the parent object's layout like a normal subform.

Related pages



Proxy business object

Proxy BO.

[Open](#)



Server-side rules

Rules.

[Open](#)



Data sources

Data sources.

[Open](#)

